

Progress (100%)



Item 1/20



What will be the result of executing the following code?

```
try:
    raise Exception(1,2,3)
except Exception as e:
    print(len(e.args))
```

☒ it will print

☐ it will print

☐ it will raise an unhandled exception

☐ it will print

Next →

Retake Test

Back to Summary

Progress (100%)



Item 2/20



What will be the output of the following code?

```
class A:
    X = 0
    def __init__(self, v = 0):
        self.Y = v
        A.X += v

a = A()
b = A(1)
c = A(2)
print(c.X)
```

☐ 1

☒ 3

☐ 2

☐ 0

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 3/20



What will be the result of executing the following code?

```
class A:
    v = 2

class B(A):
    v = 1

class C(B):
    pass

o = C()
print(o.v)
```

☒ it will print

☐ it will print

☐ it will print an empty line

☐ it will raise an exception

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 4/20



What will be the result of executing the following code?

```
def o(p):  
    def q():  
        return '*' * p  
    return q
```

```
r = o(1)  
s = o(2)  
print(r() + s())
```

☐ it will print *

☒ it will print ***

☐ it will print **

☐ it will print ****

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 5/20



If there is a superclass named `A` and a subclass named `B`, which one of the presented invocations should you put instead of the comment?

```
class A:
    def __init__(self):
        self.a = 1

class B(A):
    def __init__(self):
        # Put selected line here.
        self.b = 2
```

☐ `A.__init__()`

☐ `__init__()`

☒ `A.__init__(self)`

☐ `A.__init__(1)`

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 6/20



What will be the effect of running the following code?

```
class A:
    def __init__(self,v):
        self.__a = v + 1

a = A(0)
print(a.__a)
```

☐ 1

☐ 2

☒ it will raise an `AttributeError` exception

☐ 0

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 7/20



What will be the result of executing the following code?

```
class A:
    def __init__(self):
        pass

a = A(1)
print(hasattr(a, 'A'))
```

☒ it will raise an exception

☐ True

☐ False

☐ 1

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 8/20



What will be the result of executing the following code?

```
def I():  
    s = 'abcdef'  
    for c in s[::2]:  
        yield c  
for x in I():  
    print(x,end='')
```

☒ it will print

☐ it will print

☐ it will print

☐ it will print an empty line

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 9/20



What will be the result of executing the following code?

```
class A:
    def __str__(self):
        return 'a'

class B:
    def __str__(self):
        return 'b'

class C(A, B):
    pass

o = C()
print(o)
```

☐ it will print

☐ it will raise an exception

☒ it will print

☐ it will print

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 10/20

A data structure described as *LIFO* is actually a:



☒ stack

☐ tree

☐ heap

☐ list

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 11/20



What will be the output of the following code?

```
class A:
    def __init__(self,v = 1):
        self.v = v

    def set(self,v):
        self.v = v
        return v

a = A()
print(a.set(a.v + 1))
```

☐ 1

☐ 0

☒ 2

☐ 3

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 12/20



What will be the output of the following code?

```
class A:  
    A = 1  
  
print(hasattr(A, 'A'))
```

☐ 0

☐ 1

☒ True

☐ False

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 13/20



What will be the result of executing the following code?

```
class I:
    def __init__(self):
        self.s = 'abc'
        self.i = 0
    def __iter__(self):
        return self
    def __next__(self):
        if self.i == len(self.s):
            raise StopIteration
        v = self.s[self.i]
        self.i += 1
        return v
for x in I():
    print(x,end='')
```

☐ it will print

☒ it will print

☐ it will print

☐ it will print

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 14/20



What will be the result of executing the following code?

```
class A:
    def a(self):
        print('a')

class B:
    def a(self):
        print('b')

class C(B,A):
    def c(self):
        self.a()

o = C()
o.c()
```

☐ it will raise an exception

☐ it will print

☒ it will print

☐ it will print

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 15/20



What will be the result of executing the following code?

```
class A:
    pass

class B(A):
    pass

class C(B):
    pass

print(issubclass(C,A))
```

☐ it will print

☐ it will print

☒ it will print

☐ it will raise an exception

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 16/20



What will be the result of executing the following code?

```
def fun(n):  
    s = '+'  
    for i in range(n):  
        s += s  
        yield s  
  
for x in fun(2):  
    print(x, end='')
```

☐ it will print ++

☐ it will print +

☐ it will print +++

☒ it will print ++++++

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 17/20



What will be the result of executing the following code?

```
class Ex(Exception):
    def __init__(self,msg):
        Exception.__init__(self,msg + msg)
        self.args = (msg,)

try:
    raise Ex('ex')
except Ex as e:
    print(e)
except Exception as e:
    print(e)
```

☐ it will raise an unhandled exception

☐ it will print `exex`

☐ it will print an empty line

☒ it will print `ex`

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 18/20



What will be the result of executing the following code?

```
class A:
    def __str__(self):
        return 'a'

class B(A):
    def __str__(self):
        return 'b'

class C(B):
    pass

o = C()
print(o)
```

☐ it will print

☒ it will print

☐ it will raise an exception

☐ it will print

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 19/20



If the class's constructor is declared as below, which one of the assignments is valid?

```
class Class:
    def __init__(self):
        pass
```



`object = Class(self)`



`object = Class(object)`



`object = Class`



`object = Class()`

← Prev

Next →

Retake Test

Back to Summary

Progress (100%)



Item 20/20



What will be the result of executing the following code?

```
def f(x):  
    try:  
        x = x / x  
    except:  
        print("a",end='')  
    else:  
        print("b",end='')  
    finally:  
        print("c",end='')  
  
f(1)  
f(0)
```

☐ it will print

☒ it will print

☐ it will raise an unhandled exception

☐ it will print

← Prev

Retake Test

Back to Summary